An Evolutionary Multiagent System for Studying the Usability of Websites

E. Mosqueira-Rey
Department of Computer Science, University of A Coruña
Campus de Elviña,
15071, A Coruña, Spain
eduardo@udc.es

D. Alonso-Ríos
Department of Computer Science, University of A Coruña
Campus de Elviña,
15071, A Coruña, Spain
dalonso@udc.es

A. Vázquez-García
Department of Computer Science, University of A Coruña
Campus de Elviña,
15071, A Coruña, Spain
mavazquez@udc.es

B. Baldonedo del Río
Department of Computer Science, University of A Coruña
Campus de Elviña,
15071, A Coruña, Spain
belen@udc.es

A. Alonso-Betanzos
Department of Computer Science, University of A Coruña
Campus de Elviña,
15071, A Coruña, Spain
ciamparo@udc.es

V. Moret-Bonillo
Department of Computer Science, University of A Coruña
Campus de Elviña,
15071, A Coruña, Spain
civmoret@udc.es

Abstract

This paper describes an evolutionary multiagent system for the semi-automated study of the usability of websites and navigation paths. The system constructs a mental model of the users trying to reach one URL from another URL, simulates the browsing process, and analyses the web pages that make up possible paths between source and destination. It automatically makes suggestions and critiques in regard to usability aspects. Finally, it selects and suitably presents significant data in support of the human expert whose task it is to evaluate usability.

1. Introduction

Usability is defined as: "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" [1]. As indicated by this definition, usability depends on the context of use, which consists of users, tasks, equipment and environments. Usability forms part of the User-Centred Design (UCD) paradigm, which states that system design should be guided by the characteristics of users rather than by the characteristics of the system itself.

A range of evaluation techniques have been developed to assist with the development of more usable products. These measure various aspects of usability and detect specific problems. Ivory and Hearst developed a classification which distinguishes between the following types of techniques [2]: *testing* (based on empirical observation of real users interacting with the system); *inspection* (based on studies of the system by means of criteria or heuristics that identify possible problems); *inquiry* (based on questions, surveys, etc., answered by users); *analytical modelling* (based on formal models that represent the user and/or the system); and, finally, *simulation* (based on emulation of interactions between real users and the system).

Some implementations of these techniques require the direct participation of humans, although it makes sense to automate them as much as possible in order to reduce time, cost and effort. Depending on the type of automation, these techniques are classified as: *capture* (collection of usability data), *analysis* (interpreting data in order to detect usability problems), or *critique* (indications of problems and also of possible improvements). Nonetheless, most of the techniques only automate a specific part of these activities, and because critique is complicated to implement, they tend to focus more on capture and analysis.

This paper describes the implementation of a system for semi-automated study of the usability of websites. It will enable suggestions to be made about usability and a critique to be obtained in regard to website elements requiring improvement. The system performs a semi-automated study because, since usability is a highly subjective concept, the last word about this issue belongs to usability experts.

Taken into account is the fact that web interfaces have certain particularities that make them different from traditional WIMP (windows, icons, mouse and pointer) interfaces: first of all, the task structure is freer; secondly, they are more oriented to information provision; and finally, individual operations are simpler and more limited (most actions consist of clicking on a link [3]).

Our system is implemented using a framework for multiagent systems with evolutionary learning. A multiagent system is composed of a group of intelligent agents, which are systems whose aim is to achieve a series of goals in a normally dynamic and unpredictable environment. Evolutionary learning, which is based on biological models of genetics and evolution, enables the learning process in agents to be enhanced. A detailed description of the framework used can be found in [4].

An evolutionary multiagent system is suitable for conducting a study of the usability of a website, which it does by simulating website user behaviour—as evolutionary agents are capable of both learning for themselves and mimicking humans. Furthermore, the system can automatically explore the entire website, thereby partially freeing the expert from the work of searching out usability problems.

2. Background

Our system is based on the semi-automation of a combination of techniques, namely inspection, modelling and simulation. The literature refers to a number of studies on partial automation of usability analysis.

In regard to inspections, tools exist that help evaluators ensure that web pages or websites agree with a series of parameters and guidelines. Some of these tools analyse HTML coding in web pages to check for broken links, to see whether images have an associated text to describe them, etc. In some cases, critique is also partially automated.

As for analytical modelling, techniques such as PUM [5] model users by means of a program that acts as the user and which is given instructions on the operations for each task.

Finally, in regard to partial automation of simulations, there are the following works:

- Chi et al [6] developed a system for measuring how website page text helps users locate information. Commencing with an initial page, the system has as its aim to enter pages containing specific information represented by means of a string of keywords. This system maps navigation paths and calculates the proportion of users who reach their goal. The results obtained are interpreted manually by a human evaluator.
- From log file data resulting from user interaction with the system, the AMME tool [7] constructs a simulation model (a Petri network) from which it extracts information on the usability of the system.
- Other tools based on Information Processor Modelling focus on psychological principles. Cognitive architectures are used to simulate system use and predictions of usability are obtained from these simulations. For example, Web Criteria has constructed a model on the basis of automated navigation of a website which it analysed with a user model based on GOMS analysis [2].

In regard to the use of genetic algorithms for usability studies, Kasik and George [8] developed an automated capture technique to generate new data on the use of a system. Genetic algorithms simulate the behaviour of inexpert users, who learn on a trial-and-error basis.

It can be observed that most of these techniques automate capture and analysis. However, our aim is to develop a system which also automates critique, that is, indications in regard to solving usability problems detected in the analysis process.

3. System architecture

Our system is based on GAEL (Generic Agents with Evolutionary Learning), which is a framework designed for the development of evolutionary intelligent agents [4].

This framework is suitable for problems in which knowledge of the environment is minimal. We know something of the environment and of the actions we can implement there, and we obtain periodic reinforcement for the actions performed, but there is no a priori knowledge available that indicates actions as correct or incorrect, given that the conditions in the environment are variable. The framework consists of reactive agents—that is, agents with absolutely no previous knowledge of the environment—, each of which possesses a situation-action rule base.

The agents implement learning at two levels. On the first level they learn by reinforcement (learning at the individual agent level), so that when the agent receives a stimulus from the environment, the first thing it does is decide if it will exploit pre-existing rules in its rule base or explore new rules. If it decides to exploit pre-existing rules, it constructs a conflict set composed of the rules that refer to the actual state of the environment, and it chooses the best rule or a rule is chosen statistically. If it decides to explore or if the conflict set is empty, a rule is created randomly that will be valid for the current state of the environment and that is not already in the agent's knowledge base. At pre-determined intervals, the agent receives reinforcement that indicates whether the execution of a series of rules has enabled it to fulfil its aims.

On the second level, an evolutionary algorithm is executed at intervals (learning at the agent population level). Agents thus become better adapted to the environment through the application of evolutionary techniques (natural selection, crossover, mutation, etc.).

This framework can be used to solve different types of problems, as global elements are defined in a generic way. For application to a specific problem, the only elements which have to be developed are problem-specific elements, as follows:

- *Environment*. The element that generates events to which the agents react.
- *Environment state*. The identifiable state of the environment that the agent uses in order to implement pattern matching with rule antecedents.
- *Action*. An operation performed by the agent on the environment, which is the consequent of the agent's rules.

It should be borne in mind that correct functioning will require a study of the characteristics of the problem, in order to ensure fine-tuning of the parameters that control the learning mechanism. Some parameters serve to indicate the properties of the population, such as the number of agents, the maximum number of rules per agent, the number of learning cycles, the rule selection strategy during exploitation, and the probability that an agent will apply a new rule rather than an existing one. Others establish different characteristics of the genetic algorithm, such as the frequency with which it is to be applied, the proportion of agents selected for reproduction, the probability that an agent will undergo a mutation, etc.

Below we describe the application of this framework to the study of website usability.

4. System functioning

4.1. System agents

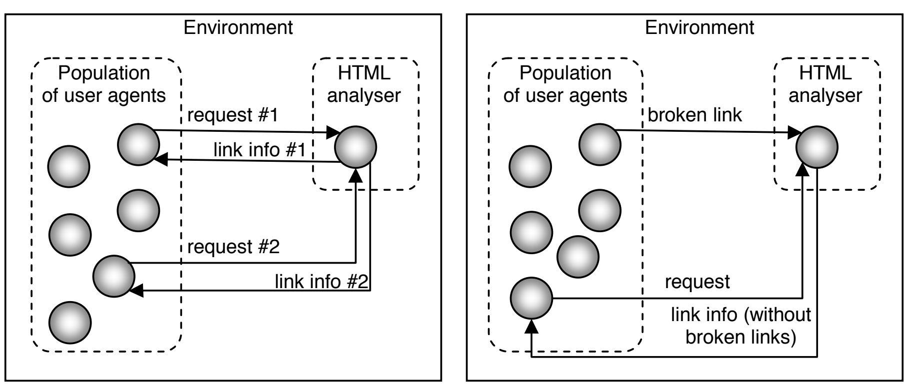
In order to apply the described multiagent architecture to a study of usability, we define the following types of intelligent agents:

1) *User agents*. Each agent has as its goal to arrive to a destination URL from an initial URL, and possesses a set of rules of potential use in achieving this goal.
2) *HTML analyser agent*. This agent does not model users; rather, it examines the web page coding and extracts useful data for the usability study. This agent can also examine the coding for an entire website.

In this domain, the problem-specific elements are as follows:

- *Environment*: the work area of the web browser.
- *Environment state*: any URL that could be visited by the user agents.
- *Action*: clicking on a link.

Our system functions on the basis of a division of labour between different specialist agents. The fact that tasks are shared out means that agents behave socially as well as autonomously, meaning that they exchange knowledge among themselves. As Figure 1a shows, when a user agent arrives to a new page, it requests information from the HTML analyser agent on the available links (link text, text surrounding the link, target URL). The user agent then checks this information against its rules. The HTML analyser agent stores the information on the links for



(a) (b)

Figure 1. Exchange of knowledge among agents. (a) User agents request information on the available links. (b) Agents share information on broken links.

future requests. Figure 1b shows another example: when a user agent clicks on a broken link, this information should be notified to other agents to avoid time wasting. One way to do this is to use the HTML analyser agent as an intermediary.

4.2. Agent parameters

User agents are not identical, but are defined by a series of parameters that model different aims and user profiles. These parameters are:

1) *Initial URL.*
2) *Destination URL*: which does not necessarily have to belong to the same domain as the initial URL.
3) *List of initial key phrases*. These represent the user's mental model of aims, which may describe concepts, actions, etc. A product with a good level of usability will mean computerised task implementation that will be as close as possible to the user's mental model (in our case, the structure of the web site and the choice of link labels should be as intuitive as possible). This set of key phrases can be obtained from the description of the functions or requirements of the product, or from surveys of users. It is also possible to create agents with no initial key phrases, in which case their aims would not be those of human users; they would, rather, be dedicated purely to exploration.
4) *Awareness of surrounding text*. Usability studies conclude that 79% of users only glance at web pages, and very few users read pages word for word [3]. When selecting links, it is typical to focus on the underlined words and to ignore the text surrounding the link.
5) *Quantity of links viewed*. Similarly to the previous comment, users tend to focus solely on the most visible sets of links.
6) *Quantity of links visited before giving up*. When a user seeks certain information, the physical or mental effort invested in the search will depend on the value assigned to achieving an aim.

4.3. Evolution and learning

User agents are subject to an evolutionary process in which the best genes (i.e., rules) are passed on to the next generation. This improves the learning process and allows the exploration of the space of possible solutions within reasonable limits. As a result, the user agents are able to model different variations of plausible executions.

Since the evolutionary process is fitness-based, we need to implement a reinforcement system that will reward or penalise agent actions. The different kinds of reinforcement are as follows:

- Positive reinforcement, on being able to use the initial key phrases.
- Positive reinforcement, for similarities between the current URL and the destination URL.
- Negative reinforcement, on being obliged to return to the previous page.

Given the characteristics of our problem, it is impossible to know with absolute certainty if, in any given moment, an agent is acting in the best possible way. Reinforcement is heuristic—that is, it does not necessarily have to reflect the best way of achieving a goal.

4.4. Usability results

Our system contributes to usability studies in two ways: first of all, for aspects of usability that can be analysed automatically, the system itself makes suggestions and implements a critique based on principles documented in the usability literature; and secondly, for aspects of usability that depend on subjective evaluation, the system assists the expert with the costly task of analysis, by means of an intelligent selection of the most relevant information presented in a way that is easily digested.

The usability analyses performed by the system are focused on the following elements:

- The paths that users will likely traverse as they try to reach their destination.
- The hypertext links that constitute that paths.
- The HTML code of the web pages.
- The text content of the web pages.

With the analysis of these elements the system looks for the presence of typical symptoms of usability problems, such as:

- The unreachability of the destination.
- Confusing navigation.
- Inappropriate link texts.
- Broken links.
- Problems with the HTML code of the web pages.
- Violations of accessibility guidelines.

Other usability problems can be identified by taking into account other type of metrics such as:

- Number of pages visited.
- Number of different paths between source and destination.
- Length of said paths.
- Number of key phrases found in the links.
- Number of key phrases found in the text surrounding the links.
- Density of text.
- Length of link text.
- Page dimensions.

5. Example

To test the prototype of our system we used the Project Gutenberg website [9], one of the oldest and best known Internet sites (Figure 2). Project Gutenberg claims to have the first and largest single collection of free electronic books. In a website of this kind, which aims to store material and receive casual visitors, navigation should be simple and intuitive—which was why this site was considered to be ideal for the first application of our system.

Our agents model users who visit the Project Gutenberg website on having learned that they can download a CD-ROM containing a collection of science fiction books. The list of key phrases selected was "download", "science fiction", "sci-fi", "sf", "cd-rom", "cd", "compact disc".

The destination URL has two features of interest: first of all, it is a *.torrent* file, and not a web page; secondly, it is not located in the Project Gutenberg domain but in a remote server.

In our first test, we decided to parameterise the system in such a way that the agents would go directly to the links and ignore the surrounding text. The outcome, even after a reasonable number of cycles, was that no agent achieved the goal, since there are no links with the given key phrases inside. We then parameterised the system so that the agents would read the surrounding text. On this occasion the agents reached the destination URL on the basis of the key phrases provided.

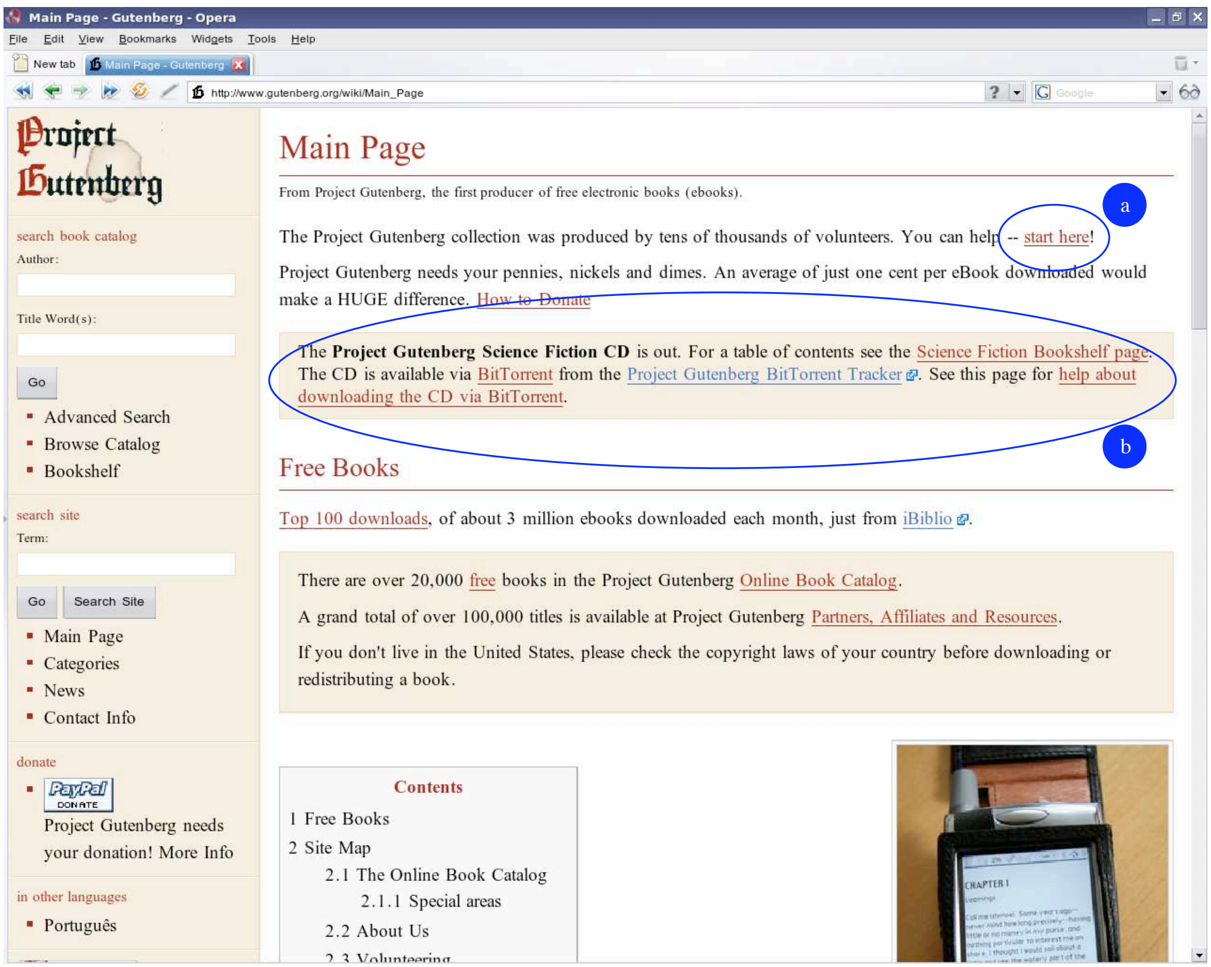


Figure 2. Project Gutenberg main page. (a) Example of inappropriate link text. (b) Text containing links to download the science fiction CD.

The minimum sequence of links in order to achieve the goal was as follows:

Project Gutenberg BitTorrent Tracker → DL (only the first link with that text that appears in the page)

Another sequence of valid links—more intuitive but longer—was as follows:

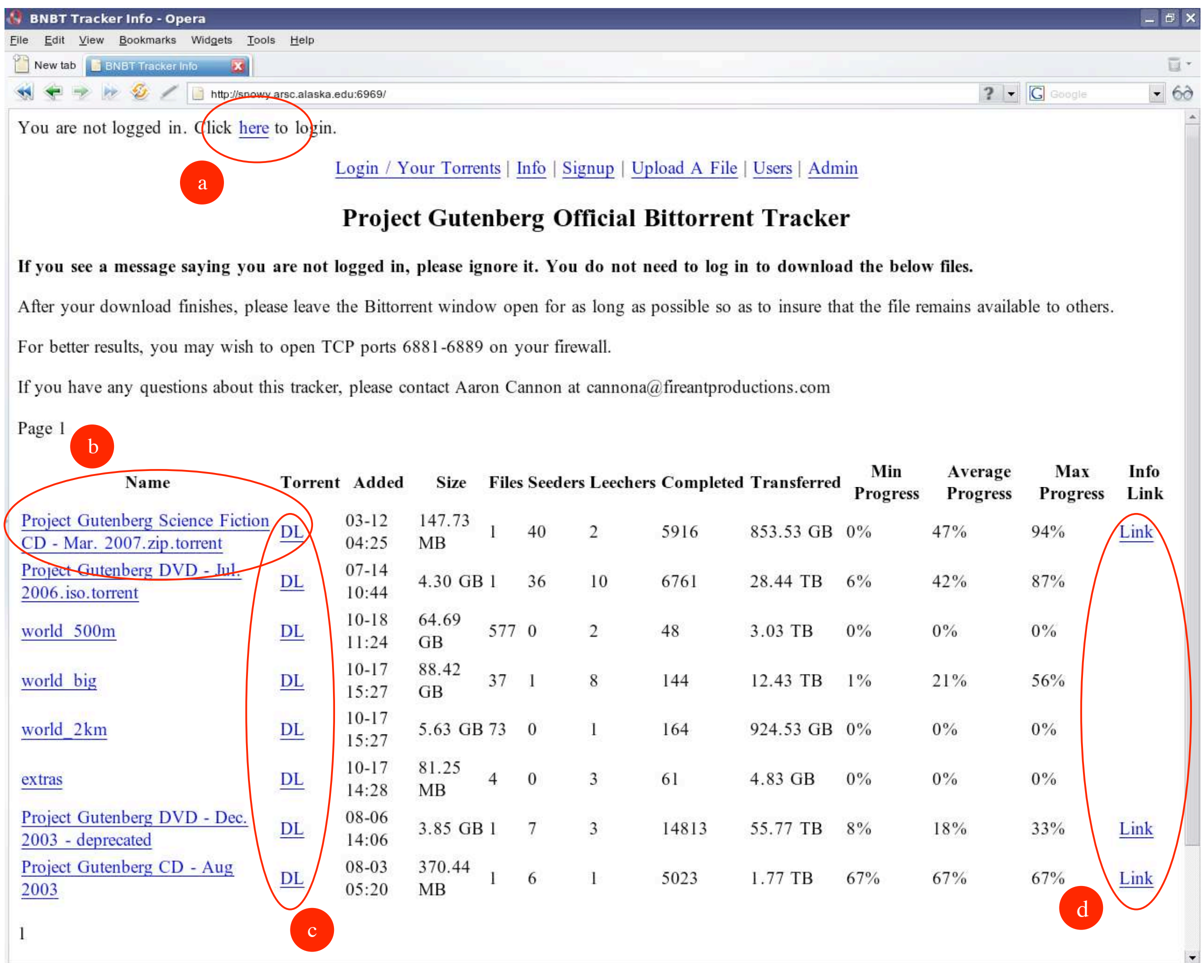
Project Gutenberg BitTorrent Tracker → Project Gutenberg Science Fiction CD - Mar. 2007.zip.torrent → DOWNLOAD TORRENT

The system automatically drew the following usability comments:

1) The destination URL can be reached by following a sequence of only two links (this is considered to be perfectly acceptable [10]).
2) There are different paths to the destination (flexibility is important for usability).
3) The pages visited are legible in a text-only browser, as all the images have an associated ALT attribute.

Additionally, the system also made the following suggestions and critiques:

1) The web pages are overly dense in terms of information (as a general rule, around half the words used in conventional writing should be used in writing web pages [3]).
2) Rewriting the labels for the links was recommended, for the following reasons:
   - The optimal link path does not contain any of the key phrases, although some of these do appear in the surrounding text (for example, see Figure 2b and Figure 3b).
   - The fact that an intermediate page contains several links with identical text ("DL", "Link") but leading



You are not logged in. Click here to login.

Login / Your Torrents | Info | Signup | Upload A File | Users | Admin

Project Gutenberg Official Bittorrent Tracker

If you see a message saying you are not logged in, please ignore it. You do not need to log in to download the below files.

After your download finishes, please leave the Bittorrent window open for as long as possible so as to insure that the file remains available to others.

For better results, you may wish to open TCP ports 6881-6889 on your firewall.

If you have any questions about this tracker, please contact Aaron Cannon at cannona@fireantproductions.com

Page 1

Name	Torrent	Added	Size	Files	Seeders	Leechers	Completed	Transferred	Min Progress	Average Progress	Max Progress	Info Link
Project Gutenberg Science Fiction CD - Mar. 2007.zip.torrent	DL	03-12 04:25	147.73 MB	1	40	2	5916	853.53 GB	0%	47%	94%	Link
Project Gutenberg DVD - Jul. 2006.iso.torrent	DL	07-14 10:44	4.30 GB	1	36	10	6761	28.44 TB	6%	42%	87%	
world_500m	DL	10-18 11:24	64.69 GB	577	0	2	48	3.03 TB	0%	0%	0%	
world_big	DL	10-17 15:27	88.42 GB	37	1	8	144	12.43 TB	1%	21%	56%	
world_2km	DL	10-17 15:27	5.63 GB	73	0	1	164	924.53 GB	0%	0%	0%	
extras	DL	10-17 14:28	81.25 MB	4	0	3	61	4.83 GB	0%	0%	0%	
Project Gutenberg DVD - Dec. 2003 - deprecated	DL	08-06 14:06	3.85 GB	1	7	3	14813	55.77 TB	8%	18%	33%	Link
Project Gutenberg CD - Aug 2003	DL	08-03 05:20	370.44 MB	1	6	1	5023	1.77 TB	67%	67%	67%	Link

1

Figure 3. Project Gutenberg BitTorrent Tracker. (a) Example of inappropriate link text. (b) Text containing links to download the CD. (c, d) Links with identical text leading to different URLs.

to different URLs might confuse navigation (see Figure 3c and Figure 3d).
- Some links for visited pages contain words that are usually indicative of inappropriate nomenclature, that is, they are computer jargon words ("http") or words that do not contribute to describing the task ("start here", "here", "link"), as Figure 2a, Figure 3a and Figure 3d show.

Observing the results of the executions modelled by the system, usability experts made the following observations on the website content semantics:

1) The way in which the website is written is not appropriate, given its readership. The links in the website focus on computerised implementation rather than on the aims of users. For example, in order to achieve their goal, users need to know what a BitTorrent Tracker is (see Figure 2b) and how it works. Furthermore, the Tracker page makes a questionable use of abbreviations: in order to be able to follow the optimal path, the user needs to infer that "DL" in this context is the abbreviation for download (see Figure 3c).
2) Project Gutenberg is available in Portuguese as well as in English, and when testing the site for this language, our experts observed that at one point the link path changed language without warning. Given that this may confuse users—they may think they did something wrong and/or they may search in vain for a language change button—, the language shift should be indicated by adding, for example, "(in English)" to the end of the label for the link that leads to a page in the other language.

6. Discussion and conclusions

This paper describes an evolutionary multiagent system for the semi-automated study of the usability of websites and navigation paths. The system is constructed on the basis of a combination of:

1) Artificial Intelligence concepts (intelligent agents, evolution).
2) Usability evaluation activities (analysis, critique).
3) Usability evaluation methods (inspection, analytical modelling, simulation).

The system is capable of automatically making suggestions on aspects of usability such as:

- The path between two URLs with the minimum number of links.
- The existence of alternative paths.
- Easily locating paths on the basis of a set of words that represent a mental model of the aims of the user.
- The length of navigation paths.
- Problems with the HTML coding for the web pages.
- Problems with the content of web pages.
- Problems with links between web pages.

Likewise, the system correctly selects and presents additional usability data with the aim of facilitating the human expert's task of evaluating website usability.

Automating evaluation tasks provides abundant information on the usability of a product both rapidly and inexpensively. The manual alternative would be a lengthy, slow and labourious process [2]. Although implementation using intelligent agents is a logical solution to automating tasks that are normally carried out by humans [11], the use of human experts continues to be necessary. First of all, because usability is relative to the context of use [1], and secondly, because intelligent agents are as yet incapable of completely replacing humans when it comes to information comprehension and disambiguation [11].

In this regard, it could be argued that identifying usability problems is similar to medical diagnosis: whereas the detection of some diseases is trivial if suitable medical instruments are available, other diseases require the opinion(s) of one or more human experts. Our multiagent system can be considered, thus, as one component in a semi-automated study of usability.

The use of reactive agents is particularly appropriate for our problem, since users who visit a website often have little previous knowledge of how the website works. It is important for the navigation of a website to be an intuitive process. Furthermore, websites generally have a freer structure than traditional interfaces [2], and this requires a more dynamic and flexible approach. The purely reactive behaviour of our agents also makes them capable of adapting automatically to changes in the environment (for example, to a restructuring of the website).

The application of evolutionary techniques to intelligent agents has two very different effects: on the one hand, it improves the learning process, and on the other hand, it enables the space of possible solutions to be explored.

Automatic inference of the list of ideal actions is useful given the typical ambiguity in website tasks: it will, moreover, act as a guide to users in managing the web application. Furthermore, we are not solely interested in success. If agents fail, it is interesting to know why—since if achieving aims is difficult or impossible, this is probably because the website has usability problems.

A combination of system parameterisation and the application of evolutionary techniques results in a controlled randomness, with agents emulating variations in human behaviour within reasonable limits. This contrasts with traditional techniques such as GOMS modelling (which is excessively rigid and limited [2]), and exhaustive or arbitrary inspections, which fail to take into account the mental model of users.

7. Acknowledgements

The research has been partly funded by the Ministerio de Educación y Ciencia (Project TIN2005-04653) and the Xunta de Galicia (Project PGIDT06PXIB105205PR).

8. References

[1] International Organization for Standardization, *ISO 9241-11:1998. Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs), Part 11: Guidance on Usability*, Geneva, Switzerland, 1998.

[2] M. Ivory, M. Hearst, "The State of the Art in Automating Usability Evaluation of User Interfaces", *ACM Computing Surveys*, vol. 33, no. 4, 2001, pp. 470-516.

[3] J. Nielsen and H. Loranger, *Prioritizing Web Usability*, New Riders Press, Berkeley, USA, 2006.

[4] E. Mosqueira-Rey, D. Alonso-Ríos, A. Vázquez-García, B. Baldonedo del Río, A. Alonso-Betanzos, V. Moret-Bonillo, "An Evolutionary Approach to Including Learning Mechanisms in Multi-Agent Systems", *Dynamics of Learning Behavior and Neuromodulation Workshop, 9th European Conference on Artificial Life (ECAL), Lisbon, Portugal, 2007*.

[5] R.M. Young, T.R.G. Green, T. Simon, "Programmable User Models for Predictive Evaluation of Interface Designs", *Proceedings of the Conference on Human Factors in Computing Systems*, ACM Press, New York, USA, 1989, pp. 15-19.

[6] E.H. Chi, P. Pirolli, J. Pitkow, "The Scent of a Site: A System for Analyzing and Predicting Information Scent, Usage, and Usability of a Website", *Proceedings of the Conference on Human Factors in Computing Systems, The Hague, The Netherlands*, ACM Press, New York, USA, 2000, pp. 161-168.

[7] M. Rauterberg, R. Aeppili, "Learning in Man-Machine Systems: The Measurement of Behavioural and Cognitive Complexity", *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, Institute of Electrical and Electronics Engineers, Vancouver, Canada, 1995, pp. 4685-4690.

[8] D.J. Kasik, and H.G. George, "Toward Automatic Generation of Novice User Test Scripts", *Proceedings of the Conference on Human Factors in Computing Systems, Vol. 1, Vancouver, Canada*, ACM Press, New York, USA, 1996, pp. 244-251.

[9] Project Gutenberg [Online document], 2007 May 4, [cited 2007 May 30], Available HTTP: http://www.gutenberg.org/

[10] J. Nielsen, *Designing Web Usability*, New Riders Publishing, Indianapolis, USA, 2000.

[11] L. Lesnick and R. Moore, *Creating Cool Intelligent Agents for the Net*, IDG Books Worldwide Inc., Foster City, USA, 1997.